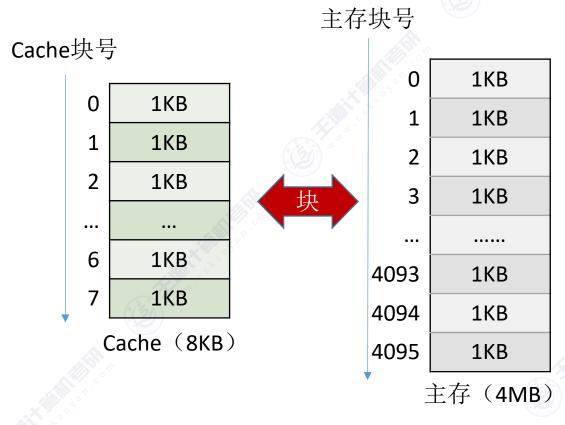
# 本节内容

# Cache-主存

映射方式

# 有待解决的问题



注意:每次被访问的主存块, 一定会被立即调入Cache

主存的地址共22位:

块号	块内地址		
12位	10位		

 $4M=2^{22}$ ,  $1K=2^{10}$ 整个主存被分为 212 = 4096 块

- 如何区分 Cache 与 主存 的数据块对应关系?
- Cache 很小,主存很大。如果Cache满了怎么办?
- CPU修改了Cache中的数据副本,如何确保主存中数据母本的一致性? ——Cache写策略
- ——Cache和主存的映射方式
- ——替换算法



还要增加"有效位"



# 本节总览



如何区分Cache中存 放的是哪个主存块?

二进制表示, 主存 主存 初始都为0 有效 标记 位 Cache Cache Cache 8 第0组 6 3 第1组 5 9 第2组 10 6 10 6 10 11 第3组 12 12 13 13 13 14 14 14 15 15 (a) 全相联映射 (b) 直接映射 (c) 组相联映射

> 主存块可以放在 Cache的任意位置

每个主存块只能放到一个特定的位置: Cache块号=主存块号% Cache总块数 Cache块分为若干组,每个主存块可放到特定分组中的任意一个位置组号=主存块号%分组数

关注公众号【研途小时】获取后续课程完整更新 !

假设某个计算机的主存地址空间大小为256MB,按**字节**编址,其数据Cache有8个Cache行,行长为64B。

即Cache块,与主存块的大小相等

256M=228 主存的地址共28位:

主存块号	块内地址		
22位	6位		

每个主存块的地址范围

	Cache	
0		40
1		,0
2		
2	(2) * ·	
4		
5 6 7		
6		
7		
		ı

Cache:  $8 \times 64B = 512B$ 

块号	主存
0	
1	(2)
2	
2 <sup>22</sup> -3 2 <sup>22</sup> -2	
$2^{22}-2$	
2 <sup>22</sup> -1	

 $0...00000000000 \sim 0...00001111111$   $0...00010000000 \sim 0...0001111111$   $0...00100000000 \sim 0...0010111111$   $1...11010000000 \sim 1...1101111111$   $1...11100000000 \sim 1...1110111111$ 

 $1...11110000000 \sim 1...11111111111$ 

假设某个计算机的主存地址空间大小为256MB,按**字节**编址,其数据Cache有8个Cache行,行长为64B。

即Cache块,与主存块的大小相等

256M=228 主存的地址共28位:

主存块号	块内地址		
22位	6位		

	Cache	
0	64B	
1	64B	
2	64B	
3	64B	以"块"为 交换单位
4	64B	父拱甲位
5	64B	
6	64B	
7	64B	
Cache	e: 8×64B =	512B

块号	主存	每个主存块的地址范围
0	64B	0000000000000000000000000000000000
1	64B	$00001000000 \sim 00001111111$
2	64B	$00010000000 \sim 00010111111$
2 <sup>22</sup> -3	64B	$11101000000 \sim 11101111111$
$2^{22}-2$	64B	$111100000000 \sim 111101111111$
2 <sup>22</sup> -1	64B	$111110000000 \sim 111111111111$

假设某个计算机的主存地址空间大小为256MB,按**字节**编址,其数据Cache有8个Cache行,行长为64B。

即Cache块,与主存块的大小相等

256M=228 主存的地址共28位:

主存块号	块内地址		
22位	6位		

每个主存块的地址范围

有效位	标记(22位)	Cache	
0	0		
0	1		
0	2		
0	3	30	
0	4		
0	5		
0	6		
0	7		
	Cach	e: $8 \times 64B = 512B$	

>\ J	٦٠,١٦	4 1 工 11 分 11 1 12 11 1 1 1 1 1
0		0000000000000000000000000000000000
1	(3	$00001000000 \sim 00001111111$
2		$00010000000 \sim 00010111111$
2 <sup>22</sup> -3		$111010000000 \sim 111011111111$
2 <sup>22</sup> -2		$111100000000 \sim 111101111111$
2 <sup>22</sup> -1		$111110000000 \sim 1111111111111$
		1

块号

主存

假设某个计算机的主存地址空间大小为256MB,按**字节**编址,其数据Cache有8个Cache行,行长为64B。

即Cache块,与主存块的大小相等

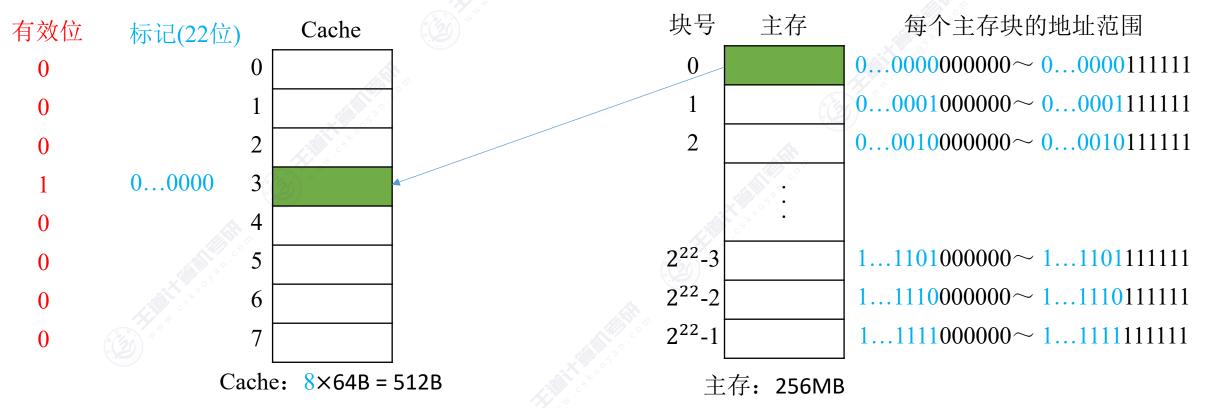
主存块号	块内地址		
22位	6位		

有效位	标记(22位)	Cache		块号	主存	每个主存块的地址范围
0	0			0		0000000000000000000000000000000000
0	1			1	Œ	$00001000000 \sim 00001111111$
0	2			2		$000100000000 \sim 000101111111$
0	3	3)			5 1 1 1 2 5 ° ° ° ° ° ° ° ° ° ° ° ° ° ° ° ° ° °	
0	4					
0	5			$2^{22}-3$		$111010000000 \sim 111011111111$
0	6			$2^{22}-2$		$111100000000 \sim 111101111111$
0	7			$2^{22}$ -1		$111110000000 \sim 111111111111$
	Cach	e: 8×64B =	512B	主	存 <b>: 2</b> 56MB	

假设某个计算机的主存地址空间大小为256MB,按**字节**编址,其数据Cache有8个Cache行,行长为64B。

即Cache块,与主存块的大小相等

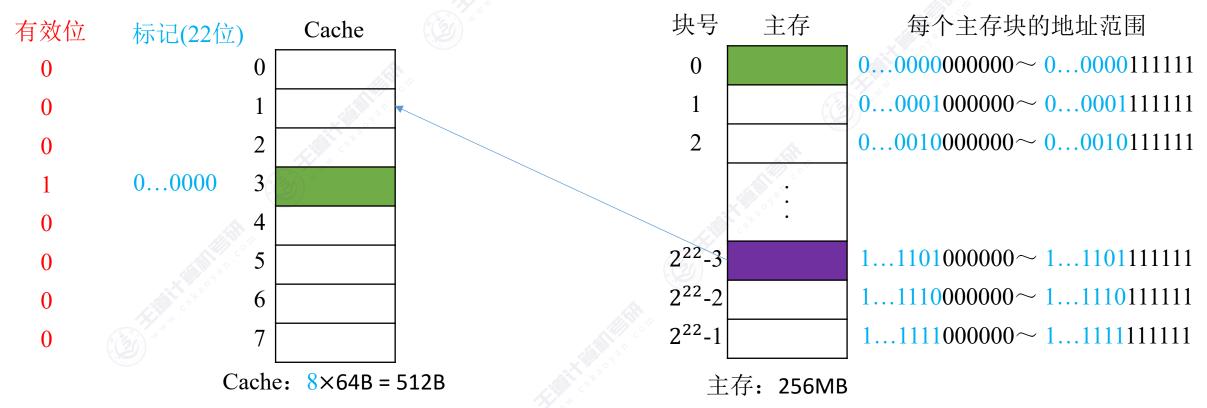
主存块号	块内地址
22位	6位



假设某个计算机的主存地址空间大小为256MB,按**字节**编址,其数据Cache有8个Cache行,行长为64B。

即Cache块,与主存块的大小相等

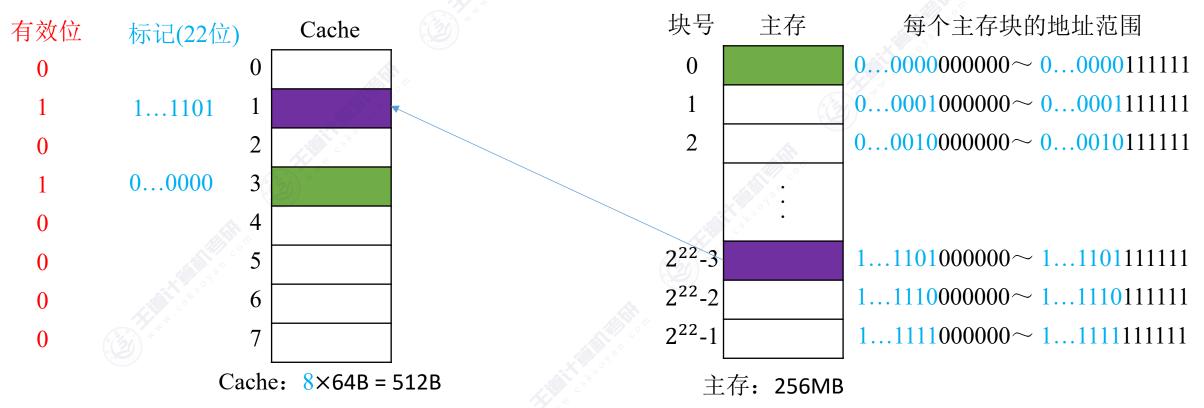
主存块号	块内地址		
22位	6位		



假设某个计算机的主存地址空间大小为256MB,按**字节**编址,其数据Cache有8个Cache行,行长为64B。

即Cache块,与主存块的大小相等

主存块号	块内地址
22位	6位



# "全相联映射"如何访存?

假设某个计算机的主存地址空间大小为256MB,按**字节**编 址,其数据Cache有8个Cache行,行长为64B。

Cache:  $8 \times 64B = 512B$ 

即Cache块,与主存块的大小相等

256M=228 主存的地址共28位:

主存块号	块内地址
22位	6位

che

CPU 访问主存地址 1...1101001110: ①主存地址的前22位, 对比Cache中所有块 的标记;

②若标记匹配且有效 位=1,则Cache命中, 访问块内地址为 001110 的单元。

③若未命中或有效位 =0,则正常访问主存

块号	主存	每个主存块的地址范围
0		0000000000000000000000000000000000
1	(3	$00001000000 \sim 00001111111$
2		$00010000000 \sim 00010111111$
$2^{22}-3$		$11101000000 \sim 11101111111$
$2^{22}-2$		$111100000000 \sim 111101111111$
2 <sup>22</sup> -1		$111110000000 \sim 1111111111111$
•		

假设某个计算机的主存地址空间大小为256MB,按字节编址,其数据Cache有8个Cache行,行长为64B。

直接映射,主存块在Cache中的<mark>位置=主存块号%Cache总块数</mark>

256M=228 主存的地址共28位:

主存块号	块内地址
22位	6位

	Cache	
0		40
1		,0
2		
3	(2) * ·	
4		
5 6 7		
6		
7		
		ı

Cache:  $8 \times 64B = 512B$ 

块号	主存	
0		
1	(2)	
2		
<i>//</i>		
2 <sup>22</sup> -3 2 <sup>22</sup> -2		
2 <sup>22</sup> -2		
2 <sup>22</sup> -1		

每个主存块的地址范围

 $0...0001000000 \sim 0...00011111111$ 

 $0...00100000000 \sim 0...00101111111$ 

 $1...1101000000 \sim 1...11011111111$   $1...1110000000 \sim 1...1110111111$   $1...11110000000 \sim 1...1111111111$ 

假设某个计算机的主存地址空间大小为256MB,按字节编址,其数据Cache有8个Cache行,行长为64B。

直接映射,主存块在Cache中的<mark>位置=主存块号%Cache总块数</mark>

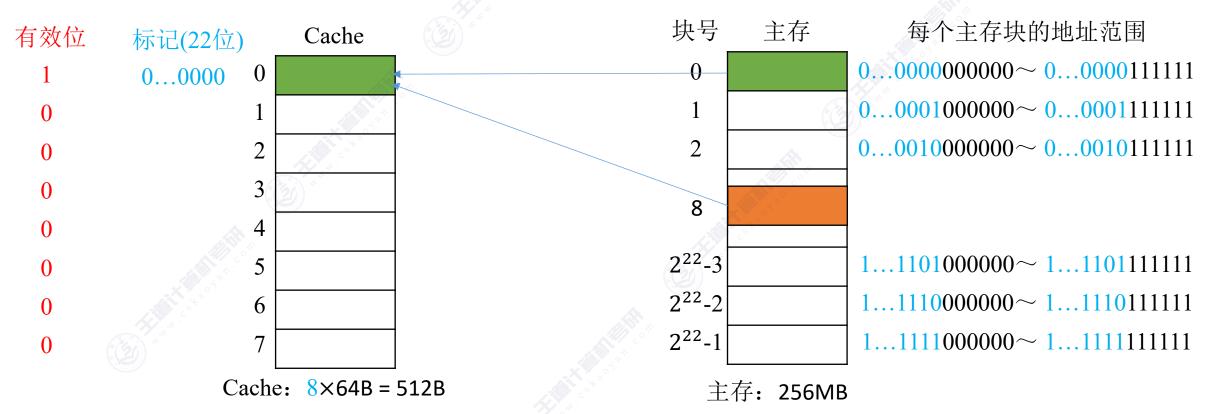
主存块号	块内地址		
22位	6位		

有效位	标记(22位)	Cache		块号_	主存	每个主存块的地址范围
1	00000 0		8	0		0000000000000000000000000000000000
0	1			1	C	$00001000000 \sim 00001111111$
0	2			2		$00010000000 \sim 00010111111$
0	3	3)				
0	4			<i>::</i> (0)		
0	5			$2^{22}-3$		$111010000000 \sim 111011111111$
0	6			$2^{22}-2$		$111100000000 \sim 111101111111$
0	7			$2^{22}-1$		$111110000000 \sim 111111111111$
	Cach	e: 8×64B =	512B	主	 存:256MB	

假设某个计算机的主存地址空间大小为256MB,按字节编址,其数据Cache有8个Cache行,行长为64B。

直接映射,主存块在Cache中的<mark>位置=主存块号%Cache总块数</mark>

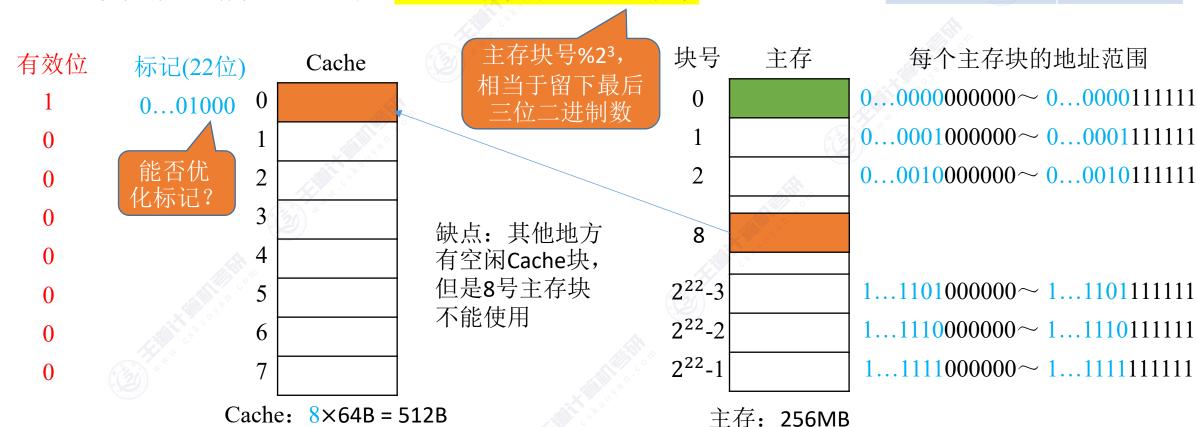
主存块号	块内地址		
22位	6位		



假设某个计算机的主存地址空间大小为256MB,按字节编址,其数据Cache有8个Cache行,行长为64B。

直接映射,主存块在Cache中的<mark>位置=主存块号%Cache总块数</mark>

主存块号	块内地址
22位	6位



假设某个计算机的主存地址空间大小为256MB,按**字节**编 址,其数据Cache有8个Cache行,行长为64B。

有效位

0

0

0

0

0

标记(22位)

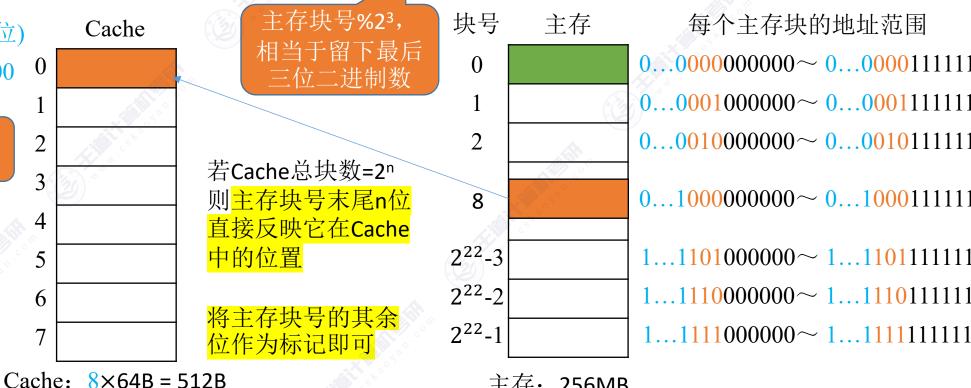
0...01000

能否优

化标记?

直接映射,主存块在Cache中的<mark>位置=主存块号%Cache总块数</mark>

每个主存块的地址范围  $0...0001000000 \sim 0...00011111111$  $0...00100000000 \sim 0...00101111111$  $0...10000000000 \sim 0...10001111111$  $1...11010000000 \sim 1...11011111111$  $1...11100000000 \sim 1...11101111111$ 



256M=228 主存的地址共28位:

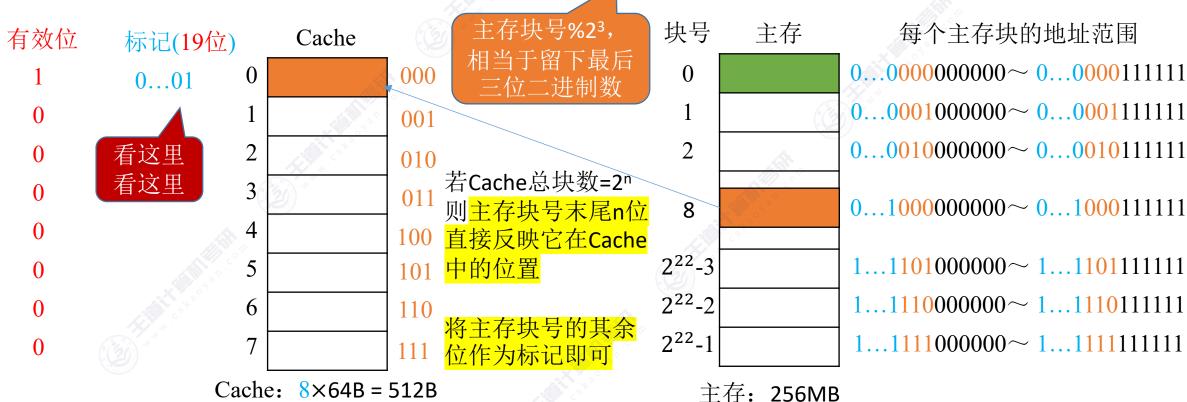
主存块号	块内地址
22位	6位

假设某个计算机的主存地址空间大小为256MB,按字节编址,其数据Cache有8个Cache行,行长为64B。

直接映射,主存块在Cache中的<mark>位置=主存块号%Cache总块数</mark>

200m 5 T.11 H1262T/(50 A.		
主存块号	块内地址	
22位	6位	
每个主存块的地址范围		

256M=228 主 存的 抽 址 共 28 位 .



# "直接映射"如何访存

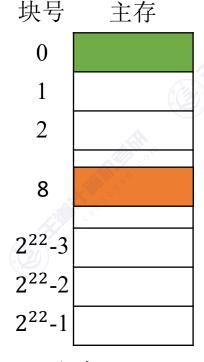
假设某个计算机的主存地址空间大小为256MB,按字节编址,其数据Cache有8个Cache行,行长为64B。

Cache:  $8 \times 64B = 512B$ 

直接映射,主存块在Cache中的<mark>位置=主存块号%Cache总块数</mark>

有效位	标记(19位)	Cache
1	001	000
0	1	001
0	2	010
0	3	011
0	4	100
0	5	101
0	6	110
0	7	111

CPU 访问主存地址
001000 001110 :
①根据主存块号的后
3位确定Cache行
②若主存块号的前19
位与Cache标记匹配
且有效位=1,则
Cache命中,访问块
内地址为 001110 的
单元。
③若未命中或有效位
=0,则正常访问主存



256M=228 主存的地址共28位:

主存	块号	块内地址
22	位	6位
<b>19</b> 位 标记	3位 行号	6位块内 地址
. 1/1 . NCT	11 7	>U >III.

#### Cache 共2<sup>3</sup> 行

每个主存块的地址范围

 $0...0001000000 \sim 0...00011111111$ 

 $0...0010000000 \sim 0...0010111111$ 

 $1...11010000000 \sim 1...11011111111$ 

 $1...11100000000 \sim 1...11101111111$ 

 $1...1111000000 \sim 1...1111111111$ 

假设某个计算机的主存地址空间大小为256MB,按字节编址,其数据Cache有8个Cache行,行长为64B。

组相联映射, 所属分组=主存块号%分组数

256M=228 主存的地址共28位:

主存块号	块内地址
22位	6位

	Cache	
0		
1		
2		
2 3 4	(2) * The state of	
4		
5		
5 6 7		
7		

Cache:  $8 \times 64B = 512B$ 

块号	主存	
0		0
1	Ć	0
2		0
	•	
2 <sup>22</sup> -3 2 <sup>22</sup> -2		1
		1
2 <sup>22</sup> -1		1

每个主存块的地址范围

 $0...00010000000 \sim 0...00011111111$ 

 $0...00100000000 \sim 0...00101111111$ 

 $1...1101000000 \sim 1...1101111111$ 

 $1...11100000000 \sim 1...11101111111$ 

 $1...11110000000 \sim 1...11111111111$ 

假设某个计算机的主存地址空间大小为256MB,按字节编址,其数据Cache有8个Cache行,行长为64B。

组相联映射, 所属分组=主存块号%分组数

256M=228 主存的地址共28位:

主存块号	块内地址
22位	6位



Cache:  $8 \times 64B = 512B$ 

 块号
 主存

 0
 1

 1
 2

 :
 :

 2<sup>22</sup>-3
 2<sup>22</sup>-2

 2<sup>22</sup>-1
 2<sup>22</sup>-1

每个主存块的地址范围

 $0...0001000000 \sim 0...00011111111$ 

 $0...00100000000 \sim 0...00101111111$ 

 $1...1101000000 \sim 1...1101111111$ 

 $1...11100000000 \sim 1...11101111111$ 

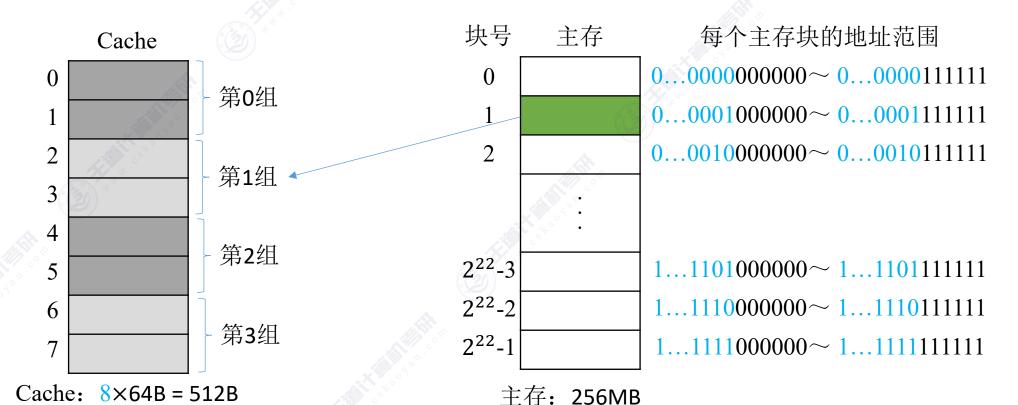
1...1111000000~1...1111111111

假设某个计算机的主存地址空间大小为256MB,按字节编址,其数据Cache有8个Cache行,行长为64B。

组相联映射, 所属分组=主存块号%分组数

256M=228 主存的地址共28位:

主存块号	块内地址
22位	6位



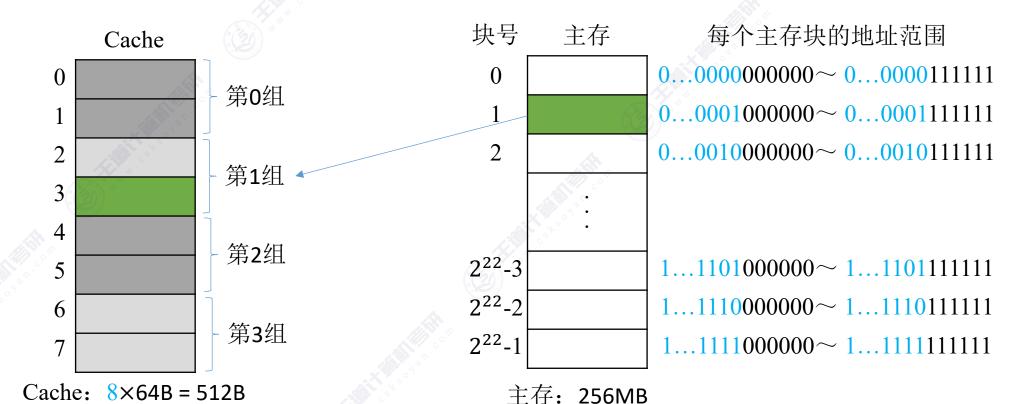
2路组相联映射——2块为一组,分<mark>图</mark>望【GIGGLOND】 获取后续课程完整更新

假设某个计算机的主存地址空间大小为256MB,按字节编址,其数据Cache有8个Cache行,行长为64B。

组相联映射, 所属分组=主存块号%分组数

256M=228 主存的地址共28位:

主存块号	块内地址
22位	6位



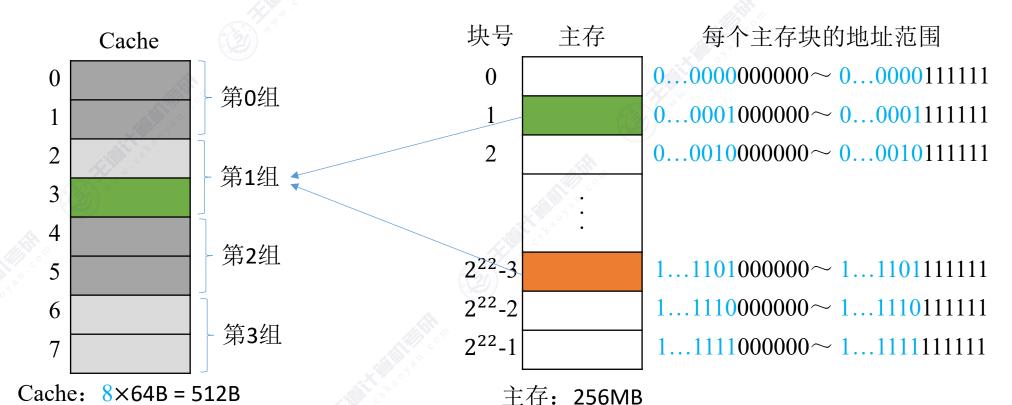
2路组相联映射——2块为一组,分<mark>图组</mark>「研途小时】获取后续课程完整更新!

假设某个计算机的主存地址空间大小为256MB,按字节编址,其数据Cache有8个Cache行,行长为64B。

组相联映射, 所属分组=主存块号%分组数

256M=228 主存的地址共28位:

主存块号	块内地址
22位	6位



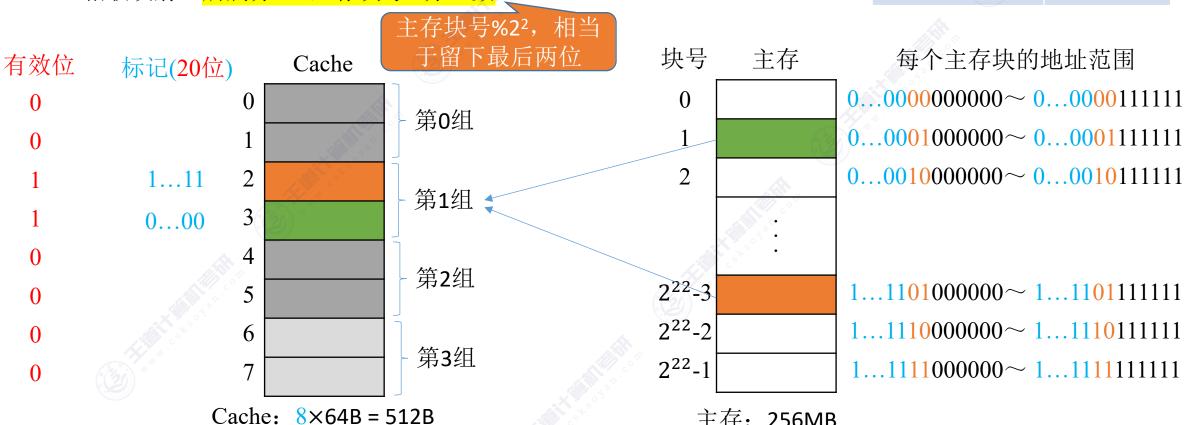
2路组相联映射——2块为一组,分<mark>图组</mark>「研途小时】获取后续课程完整更新!

假设某个计算机的主存地址空间大小为256MB,按字节编 址, 其数据Cache有8个Cache行, 行长为64B。

组相联映射, 所属分组=主存块号%分组数

256M=228 主存的地址共28位:

主存块号	块内地址
22位	6位

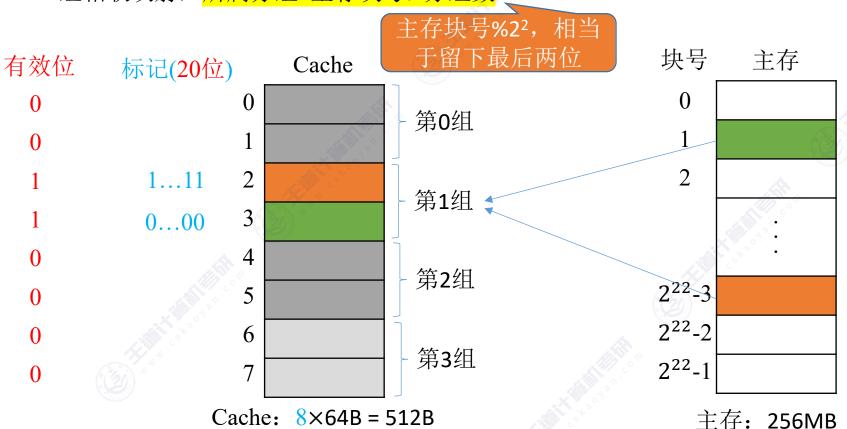


主存: 256MB

2路组相联映射——2块为一组,

假设某个计算机的主存地址空间大小为256MB,按字节编址,其数据Cache有8个Cache行,行长为64B。

组相联映射, 所属分组=主存块号%分组数



256M=228 主存的地址共28位:

主存块号		块内地址
22位		6位
<b>20</b> 位 标记	<b>2</b> 位 组号	6位块内 地址

#### Cache 分为2<sup>2</sup>组

每个主存块的地址范围

 $0...0001000000 \sim 0...00011111111$ 

 $0...0010000000 \sim 0...0010111111$ 

 $1...1101000000 \sim 1...1101111111$ 

 $1...11100000000 \sim 1...11101111111$ 

 $1...11110000000 \sim 1...1111111111$ 

# "组相联映射"如何访存

块号

2

 $2^{22}-3$ 

 $2^{22}-2$ 

 $2^{22}-1$ 

假设某个计算机的主存地址空间大小为256MB,按字节编址,其数据Cache有8个Cache行,行长为64B。

Cache:  $8 \times 64B = 512B$ 

组相联映射,所属分组=主存块号%分组数

有效位	标记(20位)		Cache
0		0	00
0		1	00
1	111	2	01
1	000	3	01
0		4	10
0		5	10
0		6	11
0		7	11

#### CPU 访问主存地址

①根据主存块号的后 2位确定所属分组号 ②若主存块号的前20 位与分组内的某个标

记匹配且有效位=1,则Cache命中,访问块内地址为001110的单元。

③若未命中或有效位 =0,则正常访问主存

#### 256M=228 主存的地址共28位:

主存块号		块内地址
22位		6位
<b>20</b> 位 标记	<b>2</b> 位 组号	6位块内 地址

#### Cache 分为2<sup>2</sup>组

每个主存块的地址范围

 $0...0001000000 \sim 0...00011111111$ 

 $0...0010000000 \sim 0...0010111111$ 

 $1...1101000000 \sim 1...1101111111$ 

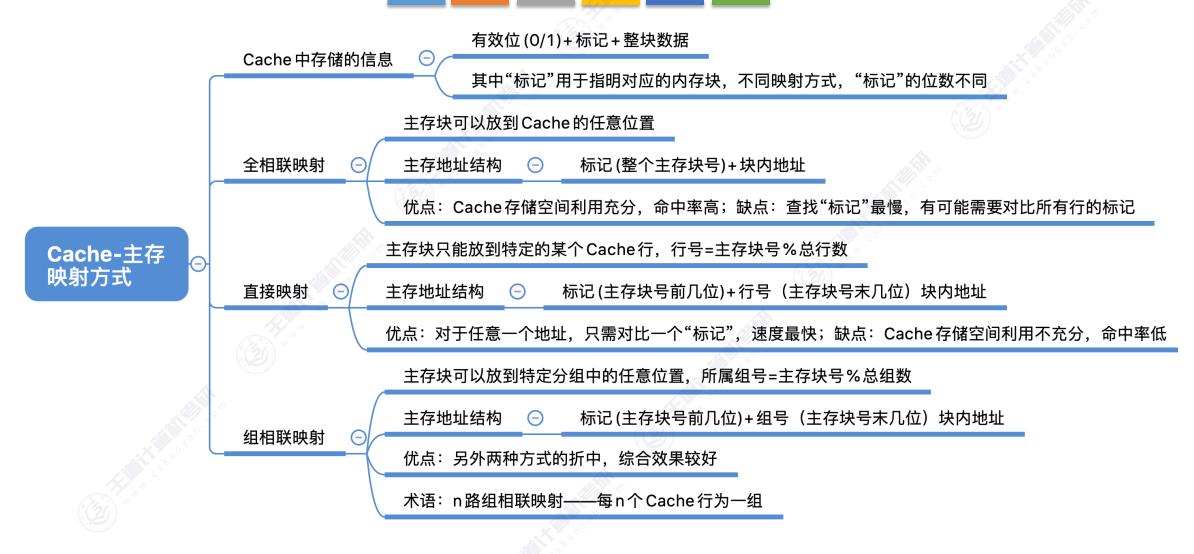
 $1...11100000000 \sim 1...11101111111$ 

 $1...1111000000 \sim 1...1111111111$ 

主存: 256MB

主存

## 知识回顾



结合每种地址映射方式的地址结构思考:给定一个主存地址,如何拆分地址,并查找Cache、访存?



△ 公众号: 王道在线



b站: 王道计算机教育



抖音:王道计算机考研